

PPPPPPPPPP
PPPPPPPPPP
PP PP PP CC LL 11 11
PPPPPPPPPP CC LL 11 11
PPPPPPPPPP CC LL 11 11
PP CCCCCCCC LLLL LLLL 111111
PP CCCCCCCC LLLL LLLL 111111

The image shows a 10x10 grid of black letters. The letters are arranged in a repeating pattern of three columns. The first column contains the letter 'L' at every position. The second column contains the letter 'I' at every position. The third column contains the letter 'S' at every position. The grid is composed of 100 letters in total.

0001
0002 C Version: 'V04-000'
0003
0004 C*****
0005 C*
0006 C* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0007 C* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0008 C* ALL RIGHTS RESERVED.
0009 C*
0010 C* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0011 C* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0012 C* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0013 C* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0014 C* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0015 C* TRANSFERRED.
0016 C*
0017 C* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0018 C* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0019 C* CORPORATION.
0020 C*
0021 C* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0022 C* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0023 C*
0024 C*
0025 C*****
0026 C
0027
0028 C Author Brian Porter Creation Date 31-MAY-1981
0029
0030
0031
0032
0033 subroutine pcl11t (lun)
0034
0035 C++
0036 C Functional description:
0037 C
0038 C This routine displays error log entries made by the
0039 C PCL11-B transmitter driver.
0040 C
0041 C Modified by:
0042 C
0043 C V03-003 SAR0229 Sharon A. Reynolds, 28-Mar-1984
0044 C Changed the call to UCBSL_OWNUIC to ORBSL_OWNER.
0045 C
0046 C V03-002 SAR0090 Sharon A. Reynolds, 20-Jun-1983
0047 C Changed the carriage control in the 'format' statements
0048 C for use with ERF.
0049 C
0050 C V03-001 SAR0041 Sharon A. Reynolds, 8-Jun-1983
0051 C Removed brief/cryptic support.
0052 C
0053 C v02-002 BP0002 Brian Porter, 23-NOV-1981
0054 C Minor edit.
0055 C
0056 C v02-001 BP0001 Brian Porter, 30-SEP-1981
0057 C Corrected call to uba_mapping.

0058 c
0059 c**
0060 c--
0061
0062 include 'src\$:msghdr.for /nolist'
0121 include 'src\$:deverr.for /nolist'
0222
0223
0224 byte lun
0225
0226 integer*4 tcr
0227 integer*4 tsr
0228 integer*4 tsdb
0229 integer*4 tsbc
0230 integer*4 tsba
0231 integer*4 tmmr
0232 integer*4 tscrc
0233 integer*4 ucb\$w_flagword
0234 integer*4 ucb\$l_lt_dpn
0235 integer*4 ucb\$l_lt_dpr
0236 integer*4 ucb\$l_lt_fmpr
0237 integer*4 ucb\$l_lt_pmp
0238 integer*4 ucb\$l_devdepend
0239
0240 equivalence (emb\$l_dv_regsav(0),tcr)
0241 equivalence (emb\$l_dv_regsav(1),tsr)
0242 equivalence (emb\$l_dv_regsav(2),tsdb)
0243 equivalence (emb\$l_dv_regsav(3),tsbc)
0244 equivalence (emb\$l_dv_regsav(4),tsba)
0245 equivalence (emb\$l_dv_regsav(5),tmmr)
0246 equivalence (emb\$l_dv_regsav(6),tscrc)
0247 equivalence (emb\$l_dv_regsav(7),ucb\$w_flagword)
0248 equivalence (emb\$l_dv_regsav(9),ucb\$l_lt_dpn)
0249 equivalence (emb\$l_dv_regsav(10),ucb\$l_lt_dpr)
0250 equivalence (emb\$l_dv_regsav(11),ucb\$l_lt_fmpr)
0251 equivalence (emb\$l_dv_regsav(12),ucb\$l_lt_pmp)
0252 equivalence (emb\$l_dv_regsav(13),ucb\$l_devdepend)
0253
0254 integer*4 compressc
0255 integer*4 compress4
0256 integer*4 destination_code
0257 integer*4 response_bits
0258 integer*4 selected_map_register
0259
0260 character*26 v1tcr(0:8)
0261 data v1tcr(0) /*START TRANSMISSION*/
0262 data v1tcr(1) /*TRANSMITTER INITIALIZE*/
0263 data v1tcr(2) /*INHIBIT ADDRESS INCREMENT*/
0264 data v1tcr(3) /*DATA OUTPUT READY*/
0265 data v1tcr(4) /*EXTENDED BUS ADDRESS 16*/
0266 data v1tcr(5) /*EXTENDED BUS ADDRESS 17*/
0267 data v1tcr(6) /*INTERRUPT ENABLE*/
0268 data v1tcr(7) /*READ SILO*/
0269
0270 character*16 v2tcr(13:15)
0271 data v2tcr(13) /*SEND WORD*/
0272 data v2tcr(14) /*TRANSMITTER NPR*/

```

0273    data      v2tcr(15)      //RETRY IF BUSY*/
0274
0275    character*22   v1tsr(4:15)
0276    data      v1tsr(4)      //TDM BUS BUSY*/
0277    data      v1tsr(5)      //SOFTWARE REJECT*/
0278    data      v1tsr(6)      //BUSY*/
0279    data      v1tsr(7)      //SUCCESSFUL TRANSFER*/
0280    data      v1tsr(8)      //DATA INPUT READY*/
0281    data      v1tsr(9)      //DATA OVERRUN*/
0282    data      v1tsr(10)     //TIMEOUT*/
0283    data      v1tsr(11)     //MASTER DOWN*/
0284    data      v1tsr(12)     //TRANSMISSION ERROR*/
0285    data      v1tsr(13)     //MEMORY OVERFLOW*/
0286    data      v1tsr(14)     //NON-EXISTENT LOCATION*/
0287    data      v1tsr(15)     //ERROR*/
0288
0289    character*29   response_a(0:3)
0290    data      response_a(0)  //TRANSMITTER ERROR/OFF-LINE*/
0291    data      response_a(1)  //NULL CYCLE*/
0292    data      response_a(2)  //VALID WORD/CRC ON DATA LINES*/
0293    data      response_a(3)  //LAST CRC ON DATA LINES*/
0294
0295    character*28   response_b(0:3)
0296    data      response_b(0)  //RECEIVER ERROR/OFF-LINE*/
0297    data      response_b(1)  //NULL CYCLE*/
0298    data      response_b(2)  //CHECK FAILURE PREVIOUS DATA*/
0299    data      response_b(3)  //ACKNOWLEDGE CRC OR DATA*/
0300
0301
0302 c++
0303 c   The format for the pcl11b transmitter buffer is as follows.
0304 c
0305 c
0306 c-----+
0307 c |          tcr
0308 c-----+
0309 c |          tsr
0310 c-----+
0311 c |          tsdb
0312 c-----+
0313 c |          tsbc
0314 c-----+
0315 c |          tsba
0316 c-----+
0317 c |          tmnr
0318 c-----+
0319 c |          tscrc
0320 c-----+
0321 c |          ucb$w_flagword
0322 c-----+
0323 c |          ucb$b_lt_retry, ucb$b_lt_maxtry
0324 c-----+
0325 c |          ucb$w_lt_dpn
0326 c-----+
0327 c |          ucb$l_lt_dpr
0328 c-----+
0329 c |          ucb$l_lt_fmpr
0330 c-----+

```

```
0330 c | ucb$1_lt_pmp  
0331 c +-----+  
0332 c | ucb$1_devdepend  
0333 c +-----+  
0334 c | ucb$w_sts  
0335 c +-----+  
0336 c--  
0337  
0338  
0339  
0340  
0341 call frctof (lun)  
0342  
0343 call dhead1 (lun,'UBA PCL11 (TRANSMITTER)')  
0344  
0345 call linchk (lun,2)  
0346  
0347 10 write(lun,10) 'TCR',tcr  
0348 format(' ',t8,a,t24,z8.4)  
0349  
0350 call output (lun,tcr,v1tcr,0,0,7,'0')  
0351  
0352 destination_code = lib$extzv (8,5,tcr)  
0353  
0354 call linchk (lun,1)  
0355  
0356 15 write(lun,15) destination_code  
0357 format(' ',t40,'DESTINATION ADDRESS :  
0358 1 i<compress4 (destination_code)>,'.')  
0359  
0360 call output (lun,tcr,v2tcr,13,13,15,'0')  
0361  
0362 call linchk (lun,1)  
0363  
0364 20 write(lun,20) 'TSR',tsr  
0365 format(' ',t8,a,t24,z8.4)  
0366  
0367 response_b ts = lib$extzv (0,2,tsr)  
0368  
0369 call lir_ck (lun,1)  
0370  
0371 25 write(lun,25) response_a(response_bits)  
0372 format(' ',t40,a<compressc (response_a(response_bits))>)  
0373  
0374 response_bits = lib$extzv (2,2,tsr)  
0375  
0376 call linchk (lun,1)  
0377  
0378 30 write(lun,30) response_b(response_bits)  
0379 format(' ',t40,a<compressc (response_b(response_bits))>)  
0380  
0381 call output (lun,tsr,v1tsr,4,4,15,'0')  
0382  
0383 call linchk (lun,1)  
0384  
0385 write(lun,20) 'TSDB',tsdb  
0386
```

```
0387 call linchk (lun,1)
0388
0389 write(lun,20) 'TSBC',tsbc
0390
0391 call linchk (lun,1)
0392
0393 write(lun,20) 'TSBA',tsba
0394
0395 call calc_map (lun,4,tcr,tsba)
0396
0397 call linchk (lun,1)
0398
0399 write(lun,20) 'TMMR',tmmr
0400
0401 call linchk (lun,1)
0402
0403 write(lun,20) 'TSCRC',tscrc
0404
0405 if (emb$w_hd_entry .ne. 98) then
0406
0407 if (ucb$l_lt_dpn .ne. 0) then
0408
0409 call uba_datapath (lun,ucb$l_lt_dpn,ucb$l_lt_dpr)
0410 endif
0411
0412 call calc_map2 (4,tcr,tsba,selected_map_register)
0413
0414 call uba_mapping (lun,selected_map_register,ucb$l_lt_fmpr)
0415
0416 if (
0417 1 lib$extzv (16,16,emb$l_dv_iosb1) .gt. 512
0418 1 .and.
0419 1 selected_map_register .ne. 0
0420 1 ) then
0421
0422 call uba_mapping (lun,(selected_map_register-1),ucb$l_lt_pmpr)
0423 endif
0424 endif
0425
0426 call linchk (lun,1)
0427
0428 write(lun,35)
0429 format(',:')
0430
0431 call orb$l_owner (lun,emb$l_dv_ownuic)
0432
0433 call ucb$l_char (lun,emb$l_dv_char)
0434
0435 call ucb$w_sts (lun,emb$w_dv_sts)
0436
0437 call ucb$l_opcnt (lun,emb$l_dv_opcnt)
0438
0439 call ucb$w_errcnt (lun,emb$w_dv_errcnt)
0440
0441 call linchk (lun,1)
0442
0443 write(lun,40) 'UCBSW_FLAGWORD',ucb$w_flagword
```

```

0444 40  format(' ',t8,a,t28,z4.4)
0445
0446  call linchk (lun,1)
0447
0448 45  write(lun,45) 'UCBSL DEVDEPEND',ucb$l_devdepend
0449  format(' ',t8,a,t24,z8.8)
0450
0451  if (emb$w_hd_entry .ne. 98) then
0452
0453  call linchk (lun,1)
0454
0455  write(lun,35)
0456
0457  call pcl11t_qio (lun,emb$w_dv_func)
0458
0459  call irp$w_bcnt (lun,emb$w_dv_bcnt)
0460
0461  call irp$w_boff (lun,emb$w_dv_boff)
0462
0463  call irp$l_pid (lun,emb$l_dv_rqid)
0464
0465  call irp$q_iosb (lun,emb$l_dv_iosb1)
0466  endif
0467
0468  return
0469  End

```

PROGRAM SECTIONS

Name	Bytes	Attributes
0 \$CODE	1287	PIC CON REL LCL SHR EXE RD NOWRT LONG
1 \$PDATA	237	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2 \$LOCAL	1432	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 EMB	512	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	3468	

ENTRY POINTS

Address	Type	Name
0-00000000		PCL11T

VARIABLES

Address	Type	Name	Address	Type	Name
2-00000308	I*4	DESTINATION_CODE	3-0000001C	L*1	EMBSB_DV_CLASS
3-00000010	L*1	EMBSB_DV_ERTCNT	3-00000011	L*1	EMBSB_DV_ERTMAX
3-0000003E	L*1	EMBSB_DV_NAMLNG	3-0000003A	L*1	EMBSB_DV_SLAVE
3-0000001D	L*1	EMBSB_DV_TYPE	3-00000036	I*4	EMBSL_DV_CHAR

3-00000012	I*4	EMBSL_DV_IOSB1
3-00000026	I*4	EMBSL_DV_MEDIA
3-0000002E	I*4	EMBSL_DV_OPCNT
3-0000001E	I*4	EMBSL_DV_RQPID
3-0000003F	CHAR	EMBST_DV_NAME
3-00000022	I*2	EMBSW_DV_BOFF
3-0000003C	I*2	EMBSW_DV_FUNC
3-0000002A	I*2	EMBSW_DV_UNIT
3-0000000E	I*2	EMBSW_HD_ERRSEQ
2-0000030C	I*4	RESPONSE_BITS
3-00000052	I*4	TCR
3-00000062	I*4	TSBA
3-0000006A	I*4	TSCRC
3-00000056	I*4	TSR
3-00000076	I*4	UCBSL_LT_DFN
3-0000007E	I*4	UCBSL_LT_FMPR
3-0000006E	I*4	UCBSW_FLAGWORD

3-00000016	I*4	EMBSL_DV_IOSB2
3-0000004E	I*4	EMBSL_DV_NUMREG
3-00000032	I*4	EMBSL_DV_OWNUIC
3-00000000	I*4	EMBSL_HD_SID
3-00000024	I*2	EMBSW_DV_BCNT
3-0000002C	I*2	EMBSW_DV_ERRCNT
3-0000001A	I*2	EMBSW_DV_STS
3-00000004	I*2	EMBSW_HD_ENTRY
AP-00000004@	L*1	LUN
2-00000310	I*4	SELECTED_MAP_REGISTER
3-00000066	I*4	TMRR
3-0000005E	I*4	TSBC
3-0000005A	I*4	TSDB
3-00000086	I*4	UCBSL_DEVDEPEND
3-0000007A	I*4	UCBSL_LT_DPR
3-00000082	I*4	UCBSL_LT_PMPR

ARRAYS

Address	Type	Name	Bytes	Dimensions
3-00000000	L*1	EMB	512	(0:511)
3-00000052	I*4	EMBSL_DV_REGSAV	420	(0:104)
3-00000006	I*4	EMBSQ_HD_TIME	8	(2)
2-00000222	CHAR	RESPONSE_A	116	(0:3)
2-00000296	CHAR	RESPONSE_B	112	(0:3)
2-00000000	CHAR	V1TCR	234	(0:8)
2-0000011A	CHAR	V1TSR	264	(4:15)
2-000000EA	CHAR	V2TCR	48	(13:15)

LABELS

Address	Label								
1-0000007A	10'	1-00000087	15'	1-000000AC	20'	1-000000B8	25'	1-000000C4	30'
1-000000D5	40'	1-000000E1	45'					1-000000D0	35'

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name	Type	Name	Type	Name	Type	Name
CALC_MAP		CALC_MAP2		I*4	COMPRESS4	I*4	COMPRESSC	Type	Name
IRPSL_PID		IRPSQ_IOSB		IRPSW_BCNT		IRPSW_BOFF		DHEAD1	FRCTOF
ORBSL_OWNER		OUTPUT		PCL11T_QIO		UBA_DATAPATH		LIBSEXTZV	LINCHK
UCBSL_OPCNT		UCBSW_ERRCNT		UCBSW_STS		UBA_MAPPING		UCBSL_CHAR	

D 5
16-Sep-1984 00:12:58
5-Sep-1984 14:20:50

VAX-11 FORTRAN V3.4-56
DISK\$VMSMASTER:[ERF.SRC]PCL11T.FOR;1

Page 8

0001
0002 Subroutine PCL11T_QIO (lun,emb\$w_dv_func)
0003
0004
0005 include 'src\$:qicommon.for /nolist'
0269
0270
0271 byte lun
0272
0273 integer*2 emb\$w_dv_func
0274
0275 integer*4 qicode(0:1,0:63)
0276
0277
0278 if (qicode(0,0) .eq. 0) then
0279
0280 qicode(1,11) = %loc(ios_writeblk)
0281
0282 qicode(1,32) = %loc(ios_writelblk)
0283
0284 qicode(1,35) = %loc(ios_setmode)
0285
0286 qicode(1,39) = %loc(ios_sensemode)
0287
0288 do 10,i = 0,63
0289
0290 qicode(0,i) = 33
0291
0292 if (qicode(1,i) .eq. 0) then
0293
0294 qicode(1,i) = %loc(qio_string)
0295 endif
0296
0297 10 continue
0298 endif
0299
0300 call irp\$w_func (lun,emb\$w_dv_func,
1 qicode(0,lib\$extzv(0,6,emb\$w_dv_func)))
0301
0302
0303 return
0304
0305 end

PROGRAM SECTIONS

Name	Bytes	Attributes
0 SCODE	126	PIC CON REL LCL SHR EXE RD NOWRT LONG
1 SPDATA	8	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2 SLOCAL	548	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 QICOMMON	1247	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	1929	

ENTRY POINTS

Address	Type	Name
0-00000000		PCL11T_QIO

VARIABLES

Address	Type	Name	Address	Type	Name
AP-000000088	I*2	EMBSW DV FUNC	2-00000200	I*4	I
3-00000442	CHAR	IOS_ABORT	3-00000340	CHAR	IOS_ACCESS
3-000003C2	CHAR	IOS_ACPCONTROL	3-000004B3	CHAR	IOS_AVAILABLE
3-00000297	CHAR	IOS_CLEAN	3-00000369	CHAR	IOS_CREATE
3-00000385	CHAR	IOS_DEACCESS	3-00000393	CHAR	IOS_DELETE
3-0000026D	CHAR	IOS_DIAGNOSE	3-00000065	CHAR	IOS_DRVCLR
3-000004CB	CHAR	IOS_DSE	3-000000A9	CHAR	IOS_ERASETAPE
3-00000276	CHAR	IOS_FORMAT	3-00000071	CHAR	IOS_INITIALIZE
3-00000014	CHAR	IOS_LOADMCODE	3-000003A1	CHAR	IOS_MODIFY
3-000003E2	CHAR	IOS_MOUNT	3-00000000	CHAR	IOS_NOP
3-0000009D	CHAR	IOS_OFFSET	3-000000EB	CHAR	IOS_PACKACK
3-000000E0	CHAR	IOS_QSTOP	3-000003EF	CHAR	IOS_RDSTATS
3-00000421	CHAR	IOS_READCSR	3-00000169	CHAR	IOS_READHEAD
3-000002B6	CHAR	IOS_READLBLK	3-0000013F	CHAR	IOS_READPBLK
3-00000200	CHAR	IOS_READRESET	3-00000195	CHAR	IOS_READTRACKD
3-0000033A	CHAR	IOS_READVBLK	3-0000045A	CHAR	IOS_READWTHBUF
3-00000484	CHAR	IOS_READWTHXBUF	3-0000004D	CHAR	IOS_RECAL
3-0000007C	CHAR	IOS_RELEASE	3-000001AB	CHAR	IOS_REREADN
3-000001B8	CHAR	IOS_REREADP	3-000000CA	CHAR	IOS_RETCENTER
3-000002E6	CHAR	IOS_REWIND	3-000002C9	CHAR	IOS_REWINDOFF
3-000000FC	CHAR	IOS_SEARCH	3-00000024	CHAR	IOS_SEEK
3-00000231	CHAR	IOS_SENSECHAR	3-00000309	CHAR	IOS_SENSEMODE
3-0000021D	CHAR	IOS_SETCHAR	3-000003B8	CHAR	IOS_SETCLOCK
3-00000088	CHAR	IOS_SETCLOCKP	3-000002DD	CHAR	IOS_SETMODE
3-000002ED	CHAR	IOS_SKIPFILE	3-000002FA	CHAR	IOS_SKIPRECORD
3-00000029	CHAR	IOS_SPACEFILE	3-0000010E	CHAR	IOS_SPACERECORD
3-000003D7	CHAR	IOS_STARTDATA	3-000000B4	CHAR	IOS_STARTDATAP
3-00000037	CHAR	IOS_STARTMPROC	3-0000020F	CHAR	IOS_STARTSPNDL
3-00000059	CHAR	IOS_STOP	3-0000000D	CHAR	IOS_UNLOAD
3-00000468	CHAR	IOS_WRITEBUFCRC	3-0000011E	CHAR	IOS_WRITECHECK
3-000001E4	CHAR	IOS_WRITECHECKH	3-000003FF	CHAR	IOS_WRITECSR
3-00000153	CHAR	IOS_WRITEHEAD	3-000002A2	CHAR	IOS_WRITELBLK
3-00000247	CHAR	IOS_WRITEMARK	3-00000314	CHAR	IOS_WRITEOF
3-0000012A	CHAR	IOS_WRITEPBLK	3-000001C9	CHAR	IOS_WITERET

PCL11T_QIO

F 5
16-Sep-1984 00:12:58
5-Sep-1984 14:20:50

VAX-11 FORTRAN V3.4-56
DISK\$VMSMASTER:[ERF.SRC]PCL11T.FOR;1

Page 10

3-0000017E CHAR IOS_WRITETRACKD
3-00000448 CHAR IOS_WRITEWTHBUF
AP-00000004a L*1 LUN

3-00000326 CHAR IOS_WRITEVBLK
3-00000257 CHAR IOS_WRTIMKR
3-000004A1 CHAR QIO_STRING

ARRAYS

Address	Type	Name	Bytes	Dimensions
2-00000000	I*4	QIICODE	512	(0:1, 0:63)

LABELS

Address	Label
**	10

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name
	IRPSW_FUNC	I*4	LIB\$EXTZV

PU

AR

LA

FU

165
15-Sep-1984 00:12:58
15-Sep-1984 14:20:50

VAX-11 FORTRAN V3.4-56

DISK\$VMSMASTER:[ERF.SRC]PCL11T.FOR;1

Page 11

0001

COMMAND QUALIFIERS

```
FORTRAN /LIS=LIS$:PCL11T/OBJ=OBJ$:PCL11T MSRC$:PCL11T
/CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)
/DEBUG=(NOSYMBOLS,TRACEBACK)
/STANDARD=(NOSYNTAX,NOSOURCE FORM)
/SHOW=(NOPREPROCESSOR,NOINCLUDE,MAP)
/F77 /NOG_FLOATING /I4 /OPTIMIZE /WARNINGS /NOD_LINES /NOCROSS_REFERENCE /NOMACHINE_CODE /CONTINUATIONS=19
```

COMPILE STATISTICS

Run Time:	5.84 seconds
Elapsed Time:	16.06 seconds
Page Faults:	192
Dynamic Memory:	193 pages

0153 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

